ARMY RESEARCH LABORATORY

The Maximin Algorithm for Adaptive Arrays and Frequency-Hopping Systems

Don Torrieri and Kesh Bakhru

ARL-TR-2026

December 1999

Approved for public release; distribution unlimited.

DTIC QUALITY INSPECTED 2

19991217 071

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

Citation of manufacturer's or trade names does not constitute an official endorsement or approval of the use thereof.

Destroy this report when it is no longer needed. Do not return it to the originator.

Army Research Laboratory

Adelphi, MD 20783-1197

ARL-TR-2026 **December 1999**

The Maximin Algorithm for Adaptive Arrays and Frequency-Hopping Systems

Don Torrieri

Information Science and Technology Directorate, ARL

Kesh Bakhru

Cubic Defense Systems

Approved for public release; distribution unlimited.

Abstract

The maximin algorithm adapts the weights of an antenna array to provide simultaneous interference suppression and beamforming in a frequency-hopping communication system. The algorithm is derived and a digital implementation is presented. Simulation experiments illustrate the ability of the maximin algorithm to rapidly form deep nulls in the directions of interference sources. For the full exploitation of an adaptive array for wideband frequency-hopping communications, some form of frequency compensation is necessary. The strengths, limitations, and variations of three methods are examined.

Preface

Recent advances in digital technology and the increased possibility of Army applications of frequency hopping have rekindled our interest in the maximin algorithm, which was originally proposed and studied more than a decade ago. This report modifies and extends the original published research results. Along with many minor improvements, there are four major revisions:

1. The theory is derived in terms of complex envelopes rather than analytic signals, because complex envelopes are directly extracted in many modern digital implementations.
2. The use of a one-sided monitor filter rather than a two-sided band-reject filter is recommended.
3. A new convergence analysis of this highly nonlinear algorithm is presented.
4. Far more extensive simulation experiments indicate the potential power and the limitations of the maximin algorithm.

Contents

Preface iii

1. Introduction 1

2. Derivation 2

3. Implementation 6

4. Convergence Analysis 10

5. Description of Simulation 14

6. Frequency Compensation 20

7. Summary 26

References 28

Distribution 29

Report Documentation Page 31

Figures

1. Basic configuration of adaptive array for frequency-hopping communications 2
2. Dehopping and initial processing 6
3. Adaptive filter that executes maximin algorithm 6
4. Transfer functions of filters for complex envelopes 7
5. (a) SINR for single simulation trial with two interference signals, each comprising tones in every frequency channel, and hopping bandwidth = 30 MHz, (b) array gain pattern at end of trial, and (c) average SINR for 20 trials 16
6. (a) SINR for single simulation trial with two interference signals, each comprising tones in 25 percent of frequency channels, and hopping bandwidth = 30 MHz, (b) array gain pattern at end of trial, and (c) average SINR for 20 trials 17
7. (a) SINR for single simulation trial with two interference signals, each comprising tones in every frequency channel, and hopping bandwidth = 300 MHz, (b) array gain pattern at end of trial, and (c) average SINR for 20 trials 20
8. Two antenna elements of adaptive array receiving plane wave 21
9. Basic configuration of adaptive array with anticipative processing 23

Tables

1. System parameters 14
2. Simulation results with edge length = λ 18
3. Simulation results with edge length = 2λ 19
4. Simulation results with edge length = λ and total hopping bandwidth = 300 MHz 25

1. Introduction

The *maximin algorithm* is an adaptive-array algorithm that suppresses interference before it enters the demodulator of a frequency-hopping communication system and thereby provides a spatial processing gain that supplements the inherent processing gain of the frequency-hopping system [1–3]. The algorithm discriminates between the desired signal and the interference on the basis of the distinct spectral characteristics of frequency-hopping signals. The maximin algorithm is so named because the desired signal is enhanced and the interference is suppressed simultaneously. The maximin algorithm is a *blind adaptive algorithm*. Blind algorithms do not require training sequences, decision-directed adaptation, or knowledge of the direction of the desired signal, but instead rely on the basic characteristics of the desired-signal waveform. The maximin algorithm requires only that the frequency-hopping pattern must be known by the receiver.

2. Derivation

The basic configuration of an adaptive array for frequency-hopping communications is displayed in figure 1. The frequency-hopping replica, which is produced by a synchronized frequency synthesizer, is used to dehop the arriving desired signals [4], thereby removing the time variations in the carrier frequency. The sampled values of the complex envelopes of the dehopped signals are extracted by the initial processors and applied to the adaptive filter, which executes the maximin algorithm.

The maximin algorithm can be derived by the method of steepest descent [4] with the signal-to-interference-plus-noise ratio (SINR) as the performance criterion. The desired signal and the interference are assumed to arrive at an adaptive array of N antennas. The desired signal, interference signals, and thermal noise are modeled as independent zero-mean, wide-sense-stationary stochastic processes. Let $\mathbf{X}(i)$ denote the discrete-time vector of the complex envelopes of the N antenna outputs after each one has been dehopped, filtered, and sampled. The index i denotes the sample number. The vector $\mathbf{X}(i)$ can be decomposed as

$$\mathbf{X}(i) = \mathbf{s}(i) + \mathbf{n}(i) \ , \tag{1}$$

where $\mathbf{s}(i)$ is the vector of desired-signal complex envelopes, and $\mathbf{n}(i)$ is the vector of interference and thermal-noise complex envelopes. Let $\mathbf{W}$ denote a vector of complex weights. Let an asterisk denote the complex conjugate, T the transpose, and H the conjugate transpose. The complex system output is

$$y(i) = \mathbf{W}^T \mathbf{X}(i) = y_s(i) + y_n(i) \ , \tag{2}$$

where

$$y_s(i) = \mathbf{W}^T \mathbf{s}(i), \quad y_n(i) = \mathbf{W}^T \mathbf{n}(i) \ . \tag{3}$$

The power in the complex signal is defined to be one-half the expected value of the magnitude squared. The factor of one-half is inserted for

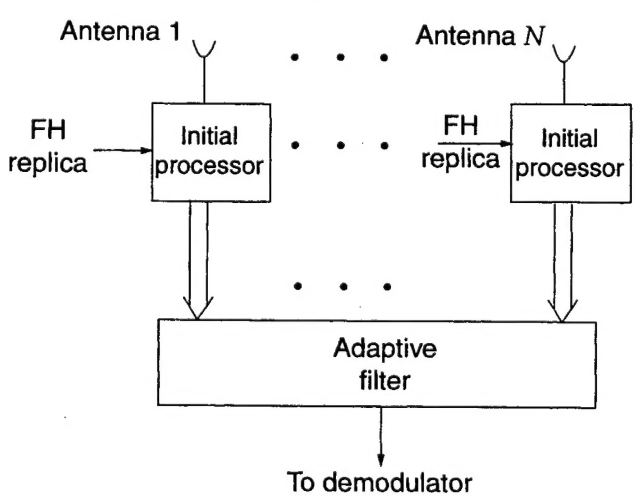


Figure 1. Basic configuration of adaptive array for frequency-hopping communications.

consistency with the convention that the power in the complex envelope $\mathbf{s}(i)$ is equal to the power in the original bandpass signal (e.g., see Proakis [5]). If $\mathbf{W}$ is regarded as deterministic, the desired-signal output power is

$$P_s = \frac{1}{2} E\left[\left| y_s(i) \right|^2\right] = \frac{1}{2} \mathbf{W}^H \mathbf{R}_{ss} \mathbf{W} \ , \tag{4}$$

where $\mathbf{R}_{ss}$ is the desired-signal correlation matrix,

$$\mathbf{R}_{ss} = E[\mathbf{s}^*(i)\ \mathbf{s}^T(i)] \ , \tag{5}$$

and $E[\mathbf{x}]$ denotes the expected value of $\mathbf{x}$. The interference-plus-noise output power is

$$P_n = \frac{1}{2} \mathbf{W}^H \mathbf{R}_{nn} \mathbf{W} \ , \tag{6}$$

where $\mathbf{R}_{nn}$ is the interference-plus-noise correlation matrix

$$\mathbf{R}_{nn} = E[\mathbf{n}^*(i)\ \mathbf{n}^T(i)] \ . \tag{7}$$

The SINR is

$$\rho = \frac{P_s}{P_n} = \frac{\mathbf{W}^H \mathbf{R}_{ss} \mathbf{W}}{\mathbf{W}^H \mathbf{R}_{nn} \mathbf{W}} \ . \tag{8}$$

To use the method of steepest descent in deriving an adaptive algorithm, it is necessary to calculate the gradient of the SINR with respect to the weight vector. In terms of its real part $\mathbf{W}_R$ and its imaginary part $\mathbf{W}_I$, a complex weight vector is defined as

$$\mathbf{W} = \mathbf{W}_R - j\mathbf{W}_I \ , \tag{9}$$

where $j = \sqrt{-1}$. Let ∇_{WR} and ∇_{WI} denote the gradients with respect to $\mathbf{W}_R$ and $\mathbf{W}_I$, respectively. The complex gradient with respect to $\mathbf{W}$ is defined as

$$\nabla_{\mathbf{w}} = \nabla_{WR} - j\nabla_{WI} \ . \tag{10}$$

A direct calculation using equations (4) and (6) gives

$$\nabla_{\mathbf{w}} \rho = \rho \left[\frac{\mathbf{R}_{ss}\mathbf{W}}{P_s} - \frac{\mathbf{R}_{nn}\mathbf{W}}{P_n} \right] \ . \tag{11}$$

Let T_s denote the time interval between samples. Let m denote the number of samples between weight iterations. The time interval between weight iterations is $m\ T_s$. The method of steepest descent for discrete-time systems gives the recursive equation for the weight vector:

$$\mathbf{W}(k+1) = \mathbf{W}(k) + \mu_0(k)\ \widehat{\nabla}(k) \tag{12}$$

where k denotes the weight iteration number, $\mu_0(k)$ is a scalar sequence that controls the rate of change in the weight vector, and $\widehat{\nabla}(k)$ is an estimate of the gradient in equation (11) at iteration k. The ideal steepest-descent algorithm results if $\widehat{\nabla}(k)$ is set equal to equation (11) with $\mathbf{W}(k)$ in place of $\mathbf{W}$. However, this algorithm requires the estimation of $\mathbf{R}_{nn}$, which is unknown in general, and $\mathbf{R}_{ss}$, which is unknown in the absence of information about the direction of the desired signal. To avoid the direct

estimation of these two matrices, several approaches are possible. If the desired signal is assumed to be narrowband, then an approximate linearization [4] of equation (11) leads to the Howells-Applebaum algorithm. Other approximations lead to the Shor feedback loop [6], which does not completely specify an adaptive algorithm. A third approach leads to the maximin algorithm, which is developed here.

The discrete-time vector $\mathbf{s}(i)$ is obtained from a continuous-time vector $\mathbf{S}(t)$ of complex envelopes. Each component of $\mathbf{S}(t)$ can be decomposed as

$$S_i(t) = S_{iR}(t) + j\, S_{iI}(t), \quad i = 1, 2, \ldots, N \ , \tag{13}$$

where $S_{iR}(t)$ is the real part or in-phase component of $S_i(t)$, and $S_{iI}(t)$ is the imaginary part or quadrature component of $S_i(t)$. Each $S_{iI}(t)$, $i = 1, 2, \ldots, N$, is a delayed version of the first component $S_{1I}(t)$, which is modeled as a zero-mean, wide-sense stationary process. Two fundamental properties of the components of a complex envelope are [4,5]

$$E\left[S_{1R}(t)\, S_{1R}(t+\tau)\right] = E\left[S_{1I}(t)\, S_{1I}(t+\tau)\right] \tag{14}$$

$$E\left[S_{1R}(t)\, S_{1I}(t+\tau)\right] = -E\left[S_{1I}(t)\, S_{1R}(t+\tau)\right] \ , \tag{15}$$

where τ is an arbitrary delay. Using these equations, we obtain

$$\mathrm{E}[\mathbf{s}(i)\, \mathbf{s}^T(i)] = \mathbf{0} \ . \tag{16}$$

The desired-signal component of the adaptive-filter output can be decomposed as

$$y_s(i) = y_{sR}(i) + j\, y_{sI}(i) \ , \tag{17}$$

where $y_{sR}(i)$ and $y_{sI}(i)$ are the real and imaginary parts of $y_s(i)$, respectively. If the weight vector is treated as a constant $\mathbf{W}$, then equations (3), (16), and (17) imply that

$$E\left[\mathbf{s}^*(i)\, y_{sR}(i)\right] = E\left[\mathbf{s}^*(i)\left\{\frac{1}{2}\mathbf{s}^T(i)\mathbf{W} + \frac{1}{2}\mathbf{s}^H(i)\mathbf{W}^*\right\}\right] = \frac{1}{2}E\left[\mathbf{s}^*(i)\,\mathbf{s}^T(i)\right]\mathbf{W} \ . \tag{18}$$

This equation and equation (5) yield

$$\mathbf{R}_{ss}\,\mathbf{W} = 2\, E[\mathbf{s}^*(i)\, y_{sR}(i)] \ . \tag{19}$$

Similarly, if the interference arriving at each antenna is a delayed version of the interference arriving at antenna 1, and if the thermal noise is independent in each array branch, then

$$E[\mathbf{n}(i)\, \mathbf{n}^T(i)] = \mathbf{0} \ , \tag{20}$$

and, hence,

$$\mathbf{R}_{nn}\,\mathbf{W} = 2\, E[\mathbf{n}^*(i)\, y_{nR}(i)] \ , \tag{21}$$

where $y_{nR}(i)$ is the real part of $y_n(i)$. Substitution of equations (19) and (21) into equation (11) yields

$$\nabla_{\mathbf{w}}\rho = 2\,\rho\left\{\frac{E\left[\mathbf{s}^* y_{sR}\right]}{P_s} - \frac{E\left[\mathbf{n}^* y_{nR}\right]}{P_n}\right\}, \tag{22}$$

where the time index has been omitted, because stationarity implies that the expected values are independent of time.

To derive the maximin algorithm, let $\hat{P}_s(k)$ and $\hat{P}_n(k)$ denote estimates of P_s and P_n at iteration k. The estimate of r at iteration k is

$$\hat{\rho}(k) = \frac{\hat{P}_s(k)}{\hat{P}_n(k)} . \tag{23}$$

Let $\overline{\mathbf{s}^* y_{sR}}(k)$ and $\overline{\mathbf{n}^* y_{nR}}(k)$ denote estimates at iteration k of the desired-signal correlation vector $E[\mathbf{s}^* y_{sR}]$ and the interference-plus-noise correlation vector $E[\mathbf{n}^* y_{nR}]$, respectively. The scalar sequence $\mu_0(k)$ in equation (12) ultimately controls the stability and rate of convergence of the maximin algorithm. It is desirable that this sequence be chosen so that as $\hat{\rho}(k)$ increases, the changes in $\mathbf{W}(k)$ will be small. Thus, we set

$$2\mu_0(k)\hat{\rho}(k) = \frac{\alpha}{\hat{\rho}(k)} , \tag{24}$$

where α is a constant. Using equations (22) and (24) and the preceding estimates in equation (12), we obtain the basic form of the maximin algorithm:

$$\mathbf{W}(k+1) = \mathbf{W}(k) + \frac{\alpha}{\hat{\rho}(k)}\left[\frac{\overline{\mathbf{s}^* y_{sR}}(k)}{\hat{P}_s(k)} - \frac{\overline{\mathbf{n}^* y_{nR}}(k)}{\hat{P}_n(k)}\right], \tag{25}$$

where $k \geq 0$ and $\mathbf{W}(0)$ is a deterministic initial weight vector. The subsequent convergence analysis and simulation results confirm that the choice given by equation (24) is effective and robust, provided that the *adaptation constant* α is appropriately selected. The first term within the brackets in equation (25) can be interpreted as a *signal term* that enables the algorithm to direct the array beam toward the desired signal. The second term within the brackets is a *noise term* that enables the algorithm to null interference signals.

The remaining issue is the choice of estimates for $\overline{\mathbf{s}^* y_{sR}}(k)$, $\overline{\mathbf{n}^* y_{nR}}(k)$, $\hat{P}_s(k)$, and $\hat{P}_n(k)$. The specific nature of frequency-hopping signals allows these estimates to be made without depending on known steering vectors or reference signals, as shown in the next section.

3. Implementation

Figure 2 illustrates the principal components of a digital implementation of the processing behind antenna 1 of an adaptive array; the processing behind the other antennas is similar. The front-end devices include a bandpass filter that excludes noise outside the hopping band and a low-noise amplifier. After dehopping from a variable frequency to a fixed intermediate frequency (IF), the received signal is filtered to prevent aliasing and to reject noise. The signal is then sampled by an analog-to-digital (A/D) converter at a rate typically four times the intermediate frequency. These samples are applied to a baseband converter, which typically includes a Hilbert transformer and a complex multiplier [7], to produce sampled values of the complex envelope of the intermediate-frequency signal. The discrete-time output of the converter is passed through a baseband filter with a bandwidth equal to that of a frequency channel. The filter blocks as much interference and noise as possible without distorting the desired signal. Thus, the passband of the frequency channel being used by the frequency-hopping desired signal is translated to baseband. The baseband-filter output is the first complex component of $\mathbf{X}(i)$, a vector applied to the adaptive filter.

Figure 3 illustrates the digital processing required by the maximin algorithm. If the interference occupies only a small fraction of the hopping

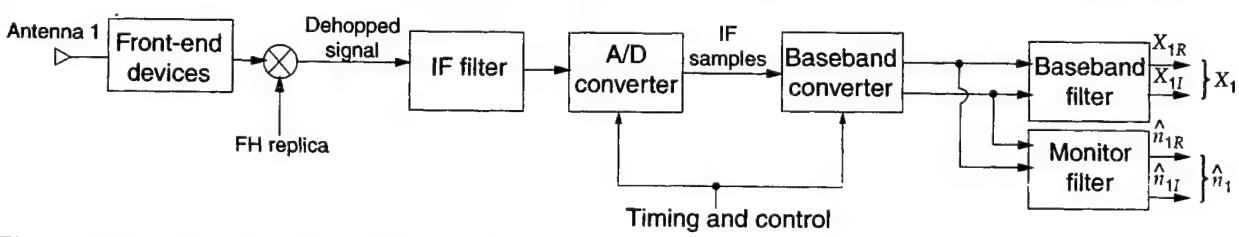
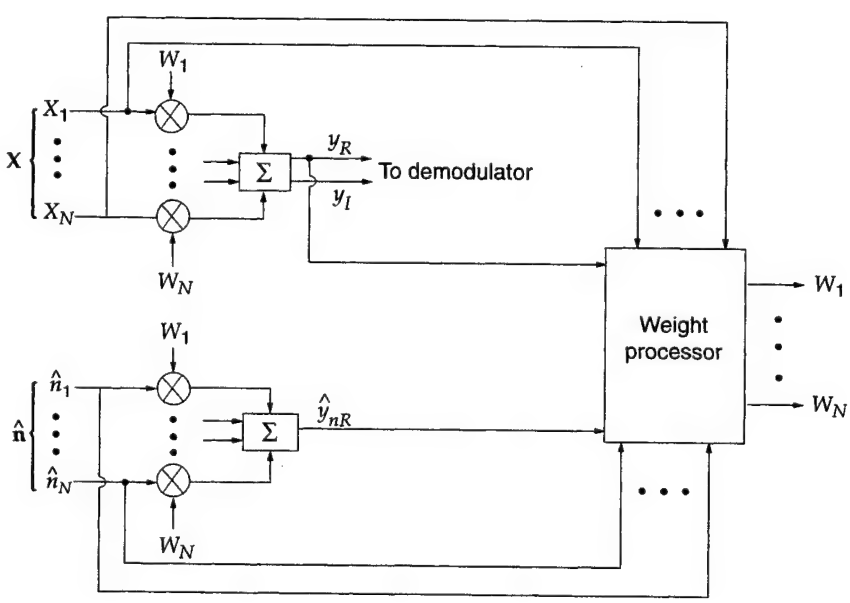


Figure 2. Dehopping and initial processing.

Figure 3. Adaptive filter that executes maximin algorithm.



band, then it is usually absent from $\mathbf{X}(i)$ and $y_R(i)$. As the weights converge to their steady-state values, the interference component of $y_R(i)$ becomes small, even if the interference is present in $\mathbf{X}(i)$. Thus, if the noise power is much less than the desired-signal power, then a suitable estimate of the desired-signal correlation vector is

$$\overline{\mathbf{s}^* y_{sR}}(k) = \frac{1}{m} \sum_{i=(k-1)m+1}^{km} \mathbf{X}^*(i)\, y_R(i) \ , \tag{26}$$

where $y_R(i)$ is the real part of the adaptive-filter output $y(i)$ given in equation (2).

An analogous estimate for $\overline{\mathbf{n}^* y_{nR}}(k)$ is not possible, because when the system hops into a part of the band with interference power, then both the desired signal and the interference are imbedded in $\mathbf{X}(i)$. Instead, the receiver observes the interference and noise in a nearby frequency channel that is not currently being used by the desired signal, but will be after a subsequent frequency hop. After a frequency translation, the interference and noise in the nearby monitored channel are extracted by each antenna's monitor filter from the baseband converter output, as shown in figure 2. The intermediate-frequency filter must have a passband large enough to encompass both the downconverted frequency channel being used by the desired signal and the downconverted monitored channel.

The monitor filter has a passband offset by f_o from the baseband filter and coinciding with a downconverted frequency channel, as depicted by the transfer function sketched in figure 4. The parameter β is the ratio of the bandwidth of the monitor filter to the bandwidth of the baseband filter. Alternatively, the monitor filter may be a band-reject filter with two passbands symmetrically located and an approximate notch over the passband of the baseband filter [1–3]. The monitor filter allows the adaptive filter to monitor the interference that will be present in the baseband filter when the carrier frequency of the desired signal eventually coincides with that of the interference. Whenever this interference is observed at the output of the monitor filter, the adaptive weights are changed to cancel the interference. There is negligible spillover of the desired signal into the output of the monitor filter if the filter offset is $f_o = 2B$ or $f_o = -2B$, where B is the bandwidth of both the baseband filter and a frequency channel. Subsequently, the single-sided monitor filter of

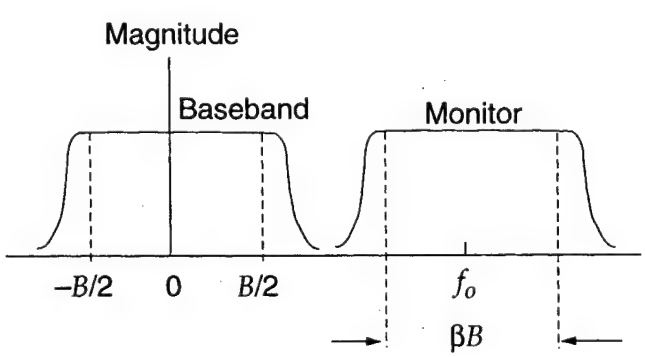


Figure 4. Transfer functions of filters for complex envelopes.

figure 4 with $\beta = 1$ is assumed, because then the monitor filter passes almost exactly the same interference waveforms as are subsequently passed by the baseband filter.

When the carrier frequency is at or near one of the ends of the hopping band, the monitor filter will not be able to observe interference in the hopping band. However, this effect will be of negligible importance when there are a large number of carrier frequencies.

Let $\hat{\mathbf{n}}(i)$ denote the vector of N discrete-time outputs of the monitor filters and define

$$\hat{y}_n(i) = \mathbf{W}^T(i)\, \hat{\mathbf{n}}(i) \ . \tag{27}$$

The vector

$$\mathbf{n}_e(i) = \hat{\mathbf{n}}(i) \exp\left(-j\, 2\, \pi\, f_o\, T_s\right) \tag{28}$$

provides an estimate of the baseband-filter input $\mathbf{n}(i)$ that will occur when the desired-signal carrier frequency is such that the interference and noise currently in the monitor filter enter the baseband filter. Treating the weight vector as a constant so that equation (21) is valid and assuming that $\mathbf{R}_{nn} \approx E\left[\mathbf{n}_e^*(i)\, \mathbf{n}_e^T(i)\right]$, we obtain

$$E\left[\mathbf{n}^*(i)\, y_{nR}(i)\right] \approx \frac{1}{2}\, E\left[\mathbf{n}_e^*(i)\, \mathbf{n}_e^T(i)\right] \mathbf{W}(i) \ . \tag{29}$$

From equation (20), it is reasonable to assume that $E\left[\mathbf{n}_e(i)\, \mathbf{n}_e^T(i)\right] \approx \mathbf{0}$. This assumption and equations (27) to (29) yield

$$\begin{aligned} E\left[\mathbf{n}^*(i)\, y_{nR}(i)\right] &\approx E\left[\hat{\mathbf{n}}^*(i)\left\{\frac{1}{2}\, \hat{\mathbf{n}}^T(i)\, \mathbf{W}(i) + \frac{1}{2}\, \hat{\mathbf{n}}^H(i)\, \mathbf{W}^*(i)\right\}\right] \\ &= E\left[\hat{\mathbf{n}}^*(i)\, \hat{y}_{nR}(i)\right] \ , \end{aligned} \tag{30}$$

where $\hat{y}_{nR}(i)$ is the real part of $\hat{y}_n(i)$. This equation implies that a suitable estimate is

$$\overline{\mathbf{n}^* y_{nR}}(k) = \frac{1}{m} \sum_{i=(k-1)m+1}^{km} \hat{\mathbf{n}}^*(i)\, \hat{y}_{nR}(i) \ . \tag{31}$$

The estimate $\hat{P}_s(k)$ can be simplified by observing from a straightforward calculation using equations (3), (16), and (17) that $E\left[y_{sR}^2\right] = E\left[y_{sI}^2\right]$, so that equation (4) implies that $P_s = E\left[y_{sR}^2\right] \approx E\left[y_R^2\right]$. Similarly, $P_n = E\left[y_{nR}^2\right] \approx E\left[\hat{y}_{nR}^2\right]$. Therefore, suitable power estimates are obtained by computing

$$\hat{P}_s(k) = \frac{1}{m} \sum_{i=(k-1)m+1}^{km} y_R^2(i) \ , \tag{32}$$

$$\hat{P}_n(k) = \frac{1}{m} \sum_{i=(k-1)m+1}^{km} \hat{y}_{nR}^2(i) \ . \tag{33}$$

In applications where the desired signal may sometimes be absent, it may be necessary to lower-bound $\hat{P}_s(k)$ by a small positive number chosen to prevent a possible singularity in the maximin algorithm.

The estimates in equations (26), (31), (32), and (33) complete the specification of the maximin algorithm given by equation (25). If the underlying assumptions are changed, then different versions of these estimates, and hence of the maximin algorithm, result. For example, if the interference occupies large spectral regions of the hopping band, then usually $y_R(i)$ will contain significant interference power before the weights converge. To compensate for this power, closely related alternative versions of the maximin algorithm can be derived [1–3]. However, many simulation experiments indicate that the version of the algorithm derived above is the most flexible, nearly always gives a similar or better performance than alternative versions, and is computationally the simplest.

During acquisition of the frequency-hopping pattern, the frequency synthesizer produces different frequencies from those of the received frequency-hopping pulses. Consequently, the desired signal is usually absent from the outputs of the baseband filters, and the maximin algorithm must be modified. During acquisition, a suitable adaptive algorithm is the recursive suppression algorithm [8], which tends to suppress signals arriving from any direction. Cancellation of the desired signal is avoided because of its infrequent appearance in the outputs of the baseband and monitor filters.

During acquisition, the adaptive system forms nulls in the directions of the interference signals, thereby assisting the receiver in synchronizing with the received frequency-hopping signal. Although a grating null might be inadvertently placed on the desired signal during acquisition, the probability of this event can usually be kept small by appropriate placement of the antenna elements [6]. After acquisition is confirmed by the receiver, it activates the maximin algorithm.

4. Convergence Analysis

A frequency-hopping signal consisting of pulses transmitted at many different frequencies is a wideband signal. However, the frequency channel associated with a single pulse is usually narrow. Thus, although a frequency-hopping signal may hop over a wide band, it has a narrow instantaneous bandwidth. For a narrowband desired signal, the vector of sampled complex envelopes is well approximated by [4,6]

$$\mathbf{s}(i) = s_1(i)\,\mathbf{S}_0 \ , \tag{34}$$

where $s_1(i)$ is the sampled complex envelope at antenna 1, which serves as a reference point, and $\mathbf{S}_0$ is a steering vector. If the antenna patterns are identical and the antennas are close enough that the signal amplitudes at all the antennas are nearly the same, then component i of the steering vector is $\exp(-j\,2\,\pi f_c\,T_i)$, where f_c is the center frequency and T_i, $2 \le i \le N$, is the arrival-time delay at antenna i relative to antenna 1. After the substitution of equations (5) and (34) into equation (8), the maximization of the SINR yields the optimal weight vector [4,6]

$$\mathbf{W}_0 = \eta \mathbf{R}_{nn}^{-1}\,\mathbf{S}_0^* \ , \tag{35}$$

where η is an arbitrary constant, and the superscript –1 denotes the inverse. For those pulses that hop over a sufficiently small band within the hopping band, the optimal weight vector is well approximated by equation (35), with $\mathbf{S}_0$ evaluated at the center frequency of the small band.

The highly nonlinear nature of the maximin algorithm precludes a completely rigorous convergence analysis. However, with enough approximations, the convergence of the mean weight vector to $\mathbf{W}_0$ can be demonstrated and bounds on the adaptation constant can be derived. Assume that enough algorithm iterations have occurred that the power estimates are well approximated by the values they would have if the weight vector were equal to $\mathbf{W}_0$. The desired-signal power at an antenna is

$$P_{si} = \frac{1}{2}\,E\left[\left|s_1(i)\right|^2\right] \ . \tag{36}$$

Using equations (4), (5), (34), (35), and (36), and the fact that $\mathbf{R}_{nn}$ is Hermitian, we obtain

$$\hat{P}_s(k) \approx P_{so} = \frac{1}{2}\,\mathbf{W}_0^H\,\mathbf{R}_{ss}\,\mathbf{W}_0 = |\eta|^2\,P_{si}\left(\mathbf{S}_0^T\,\mathbf{R}_{nn}^{-1}\,\mathbf{S}_0^*\right)^2 \ . \tag{37}$$

Similarly,

$$\hat{P}_n(k) \approx P_{no} = \frac{1}{2}\,\mathbf{W}_0^H\,\mathbf{R}_{nn}\,\mathbf{W}_0 = \frac{|\eta|^2}{2}\,\mathbf{S}_0^T\,\mathbf{R}_{nn}^{-1}\,\mathbf{S}_0^* \ . \tag{38}$$

Equations (37) and (38) imply that

$$\hat{\rho}(k) = \frac{\hat{P}_s(k)}{\hat{P}_n(k)} \approx \rho_o = 2\,P_{si}\,\mathbf{S}_0^T\,\mathbf{R}_{nn}^{-1}\,\mathbf{S}_0^* \ . \tag{39}$$

Assuming the zero-mean stationarity of the signals and the independence of the desired-signal and the interference-plus-noise, we obtain from equations (26), (1), and (2) that

$$\begin{aligned} E\left[\overline{\mathbf{s}^* y_{sR}}(k)\right] &= E\left[\mathbf{X}^*(i)\, y_R(i)\right] \\ &= E\left[s^*(i)\, y_{sR}(i)\right] + E\left[\mathbf{n}^*(i)\, y_{nR}(i)\right] . \end{aligned} \tag{40}$$

Equation (25) indicates that $\mathbf{W}(k+1)$ depends on $\mathbf{S}(l)$ and $\mathbf{n}(l)$ for $l \le k$, but not directly on $\mathbf{S}(k+1)$ and $\mathbf{n}(k+1)$. Therefore, we make the approximation that $\mathbf{W}(k)$ is statistically independent of $\mathbf{S}(k)$ and $\mathbf{n}(k)$. A calculation similar to equation (18) using equations (16) and (20) then yields

$$E\left[\overline{\mathbf{s}^* y_{sR}}(k)\right] \approx \frac{1}{2}\left(\mathbf{R}_{ss} + \mathbf{R}_{nn}\right) E\left[\mathbf{W}(k)\right] . \tag{41}$$

Similarly, equations (30) and (31) imply that

$$E\left[\overline{\mathbf{n}^* y_{nR}}(k)\right] \approx \frac{1}{2}\, \mathbf{R}_{nn}\, E\left[\mathbf{W}(k)\right] . \tag{42}$$

Taking the expected value of both sides of equation (25), substituting equations (37) to (42), and simplifying algebraically, we obtain

$$E\left[\mathbf{W}(k+1)\right] = \left\{\mathbf{I} - \frac{\alpha}{2\rho_o P_{so}}\left[\left(\rho_o - 1\right)\mathbf{R}_{nn} - \mathbf{R}_{ss}\right]\right\} E\left[\mathbf{W}(k)\right] , \tag{43}$$

where $k \ge 0$ and $E[\mathbf{W}(0)] = \mathbf{W}(0)$. Assuming that $\rho_o \gg 1$, we set $\rho_o - 1 \approx \rho_o$ and define

$$\mathbf{D} = \rho_o\, \mathbf{R}_{nn} - \mathbf{R}_{ss} . \tag{44}$$

We obtain the approximate result

$$E\left[\mathbf{W}(k+1)\right] = \left\{\mathbf{I} - \frac{\alpha}{2\rho_o P_{so}}\mathbf{D}\right\} E\left[\mathbf{W}(k)\right] . \tag{45}$$

A straightforward calculation using equations (5), (34), (36), and (39) yields

$$\mathbf{D}\, \mathbf{R}_{nn}^{-1}\, \mathbf{S}_0^* = \mathbf{0} . \tag{46}$$

This equation indicates that $\mathbf{R}_{nn}^{-1}\, \mathbf{S}_0^*$ is an eigenvector of $\mathbf{D}$, and the corresponding eigenvalue is 0. Since $\mathbf{D}$ is Hermitian, it has a complete set of N orthogonal eigenvectors, one of which is $\mathbf{R}_{nn}^{-1}\, \mathbf{S}_0^*$. Thus, we can make the decomposition

$$E\left[\mathbf{W}(k)\right] = \eta(k)\, \mathbf{R}_{nn}^{-1}\, \mathbf{S}_0^* + \sum_{i=1}^{N-1} a_i(k)\, \mathbf{e}_i , \tag{47}$$

where each $a_i(\mathrm{k})$ and $\eta(k)$ are scalar functions and $\mathbf{e}_i$ is one of the $N-1$ eigenvectors orthogonal to $\mathbf{R}_{nn}^{-1}\, \mathbf{S}_0^*$. Substituting equation (47) into equation (45) and using the orthogonality of the eigenvectors, we obtain

$$\eta(k+1) = \eta(k) = \eta \tag{48}$$

$$a_i(k+1) = \left(1 - \frac{\alpha}{2\rho_o P_{so}}\lambda_i\right) a_i(k), \; i = 1, 2, \ldots, N-1 , \tag{49}$$

where η is an arbitrary constant and λ_i is the eigenvalue corresponding to $\mathbf{e}_i$. Equations (48), (47), and (35) give

$$E[\mathbf{W}(k)] = \mathbf{W}_0 + \sum_{i=1}^{N-1} a_i(k)\, \mathbf{e}_i \ , \tag{50}$$

which implies that $E[\mathbf{W}(k)]$ converges to $\mathbf{W}_0$ if and only if each $a_i(k)$ converges to 0. From equation (49), it follows that

$$a_i(k) = \left(1 - \frac{\alpha}{2\rho_o P_{so}} \lambda_i\right)^k a_i(0), \quad k \geq 0 \ . \tag{51}$$

Therefore, $a_i(k)$ converges to 0 if and only if

$$\left|1 - \frac{\alpha}{2\rho_o P_{so}} \lambda_i\right| < 1 \ , \quad i = 1, 2, \ldots, N-1 \ . \tag{52}$$

As shown subsequently, each λ_i is positive. Equation (52) yields the necessary and sufficient convergence condition for the convergence of $E[\mathbf{W}(k)]$:

$$0 < \alpha < \frac{4\rho_o P_{so}}{\lambda_{\max}} \ , \tag{53}$$

where $\lambda_{\max}$ is the largest eigenvalue of $\mathbf{D}$.

It remains to show that each λ_i is positive. Since the SINR is maximized when $\mathbf{W} = \mathbf{W}_0$, it follows from equation (8) that

$$\frac{\mathbf{W}^H \mathbf{R}_{ss} \mathbf{W}}{\mathbf{W}^H \mathbf{R}_{nn} \mathbf{W}} \leq \rho_o \ , \tag{54}$$

where the inequality is strict when $\mathbf{W} \neq \mathbf{W}_0$. Combining equations (44) and (54) gives

$$\mathbf{W}^H \mathbf{D}\, \mathbf{W} \geq 0 \ , \tag{55}$$

which proves that the Hermitian matrix $\mathbf{D}$ is nonnegative definite. Hence all its eigenvalues must be nonnegative. Since only $\mathbf{W} = \mathbf{W}_0$ gives an equality in equation (55), setting $\mathbf{W} = \mathbf{e}_i$ implies that

$$\mathbf{e}_i^H \mathbf{D}\, \mathbf{e}_i > 0 \ , \quad i = 1, 2, \ldots, N-1 \ . \tag{56}$$

Therefore, $\lambda_i = 0$ is impossible and

$$\lambda_i > 0 \ , \quad i = 1, 2, \ldots, N-1 \ . \tag{57}$$

The sum of the eigenvalues of a square martrix is equal to its trace. Thus,

$$\begin{aligned} \lambda_{\max} &\leq \sum_{i=1}^{N-1} \lambda_i = tr(\mathbf{D}) = \rho_o tr(\mathbf{R}_{nn}) - tr(\mathbf{R}_{ss}) \\ &\leq \rho_o tr(\mathbf{R}_{nn}) \ , \end{aligned} \tag{58}$$

where $tr(\mathbf{A})$ denotes the trace of $\mathbf{A}$. Consequently, equation (53) implies that

$$0 < \alpha \leq \frac{4 P_{so}}{tr\, \mathbf{R}_{nn}} \tag{59}$$

is a sufficient condition (but not a necessary condition) for the convergence of the mean weight vector to the optimal weight vector.

Assume the interference-plus-noise power in each component of $\mathbf{n}(i)$ is equal to P_{ni}. Then $tr(\mathbf{R}_{nn}) = 2N\,P_{ni}$. Let $|\mathbf{z}|$ denote the Euclidean norm of a vector $\mathbf{z}$. From equations (5), (34), (36), and (37) and the Schwarz inequality, it follows that $P_{so} \le |\mathbf{W}_0|^2\,|\mathbf{S}_0|^2\,P_{si}$. If each component of $\mathbf{S}_0$ is a complex exponential, $|\mathbf{S}_0|^2 = N$ and $P_{so} \le N\,|\mathbf{W}_0|^2\,P_{si}$. Substituting $E[\mathbf{W}(0)] = \mathbf{W}(0)$ into equation (50) and using the orthogonality of the eigenvectors gives $|\mathbf{W}_0| \le |\mathbf{W}(0)|$. Therefore, the upper bound $a_u = 4\,P_{so}/tr\,\mathbf{R}_{nn}$ in equation (59) is approximately bounded by

$$\alpha_u \le 2\,|\mathbf{W}(0)|^2\,\rho_1\ , \tag{60}$$

where $\rho_1 = P_{si}/P_{ni}$ is the SINR in a single baseband-filter output prior to the adaptive filtering and equals the SINR per frequency channel at an antenna input. Since the bound in this inequality is easily calculated, it provides a useful, albeit imperfect, guide in selecting the appropriate value of the adaptation constant.

5. Description of Simulation

In the simulation, the antenna array consists of four omnidirectional antennas located at the vertices of a square. The symmetry of this architecture ensures a relatively mild dependence of the array response on the signal directions and allows full azimuthal coverage. The edge length is equal to 1 or 2 times the wavelength λ corresponding to the center frequency of the desired signal, 3 GHz. Each interference source is in the plane of the array. The desired signal arrives from a direction perpendicular to one of the edges, and this direction is defined to be 0°. All signals are assumed to arrive as plane waves. The system parameters are summarized in table 1.

The frequency-hopping signal has a randomly chosen carrier frequency within the hopping band and is modulated by binary minimum-shift keying (MSK). The sequence of data bits is randomly generated at the rate of 100 kbps. The hop dwell time is 1 ms. The thermal noise at the output of each intermediate-frequency filter following an antenna element is modeled as filtered white Gaussian noise. The ratio of the signal power to the thermal-noise power is defined relative to a single frequency channel and is set equal to 20 dB. Since the frequency-hopping signal is assumed to be noncoherent in carrier phase from one hop to the next, each frequency-hopping pulse has an initial phase that is uniformly distributed over the interval $[-\pi, \pi]$. The hopping frequencies are separated by 100 kHz and spread uniformly over the total hopping band, which occupies either 30 or 300 MHz. Thus, there are either 300 or 3,000 contiguous frequency channels. The total interference power due to all interference signals is equal to 10 times the number of frequency channels, which maintains a constant value of total interference power per frequency channel. Thus, ρ_1 is slightly less than –10 dB. Each interference signal has the same power and distributes tones in some of the frequency channels.

Table 1. System parameters.

Parameter	Value
Array antennas	4, omnidirectional, at vertices of square
Array edge length	1 or 2 wavelengths
Center frequency	3 GHz
Hop dwell time	1 ms
Data rate	100 kbps
Frequency modulation	MSK
Signal-to-noise ratio	20 dB per antenna and channel
Hopping bandwidth	30 or 300 MHz
Number of frequency channels	M = 300 or 3000
Monitor filter offset	200 kHz
Sampling rate	800 ksamples/s
Weight iterations per hop	8
Total interference-to-signal ratio	10 M
Interference type	Tones in channels
Number of hops per experiment	50

Perfect synchronization between the frequency-hopping signals at all the antenna outputs and the frequency synthesizer in the receiver is assumed. The baseband and monitor filters are modeled as digital 8-pole Butterworth filters with the 3-dB bandwidths equal to 100 kHz, the bandwidth of a frequency channel. The monitor filter has a single passband, as depicted in figure 4, with parameters $f_o = 200$ kHz and $\beta = 1$. The sampling rate of the analog-to-digital converter is 800 kilosamples per second. There are 8 weight iterations per hop.

In each of the subsequent simulation experiments, the initial weight vector is $\mathbf{W}(0) = \mathbf{W}_{\text{beam}}$, where

$$\mathbf{W}_{\text{beam}} = [\,1 \quad 1 \quad 1 \quad 1\,]\,. \tag{61}$$

This weight vector points the array beam or gain pattern in the direction of the desired signal. Thus, the subsequent simulation experiments indicate the performance of the maximin algorithm after the sudden appearance of interference from various directions. An alternative is to set $\mathbf{W}(0) = \mathbf{W}_{\text{omni}}$, where

$$\mathbf{W}_{\text{omni}} = [\,1 \quad 0 \quad 0 \quad 0\,]\,. \tag{62}$$

This weight vector corresponds to an omnidirectional gain pattern. Its use as the initial weight vector is appropriate if one wishes to assess the convergence of the maximin algorithm from a cold start. Since the algorithm then has to form a beam in the direction of the desired signal while simultaneously attempting to null the interference signals, the convergence is slowed relative to what it would be if $\mathbf{W}(0) = \mathbf{W}_{\text{beam}}$. Simulation experiments verify that the steady-state performance is not affected by the choice of $\mathbf{W}_{\text{beam}}$ or $\mathbf{W}_{\text{omni}}$ as the initial weight vector, so $\mathbf{W}_{\text{beam}}$ is always used.

Inequalities (59) and (60) indicate that $0 < \alpha \le \alpha_u$, where $\alpha_u \le 0.8$ is sufficient for convergence if the interference signals approximate stationary processes. Simulation experiments confirm that the maximin algorithm does converge to a steady state when $0 < \alpha \le 0.8$. It is found that $\alpha = 0.2$ usually provides close to the fastest convergence and the best performance when each interference signal comprises tones in every frequency channel.

An example of the results of a typical simulation trial is illustrated in figure 5(a). The *SINR* at each weight iteration is calculated by averaging the SINR over all sampling intervals from after the preceding weight iteration to just before the current weight iteration. The system parameters are array edge = λ and hopping bandwidth = 30 MHz. Two equal-power interference signals arrive from 40° and 70° relative to the desired-signal direction. Each of these signals distributes tones in all the frequency channels, and the adaptation constant is $\alpha = 0.2$. The SINR gradually increases until the adaptive system reaches a nearly steady-state condition during which the SINR fluctuates by a few decibels.

Figure 5. (a) SINR for single simulation trial with two interference signals, each comprising tones in every frequency channel, and hopping bandwidth = 30 MHz, (b) array gain pattern at end of trial, and (c) average SINR for 20 trials.

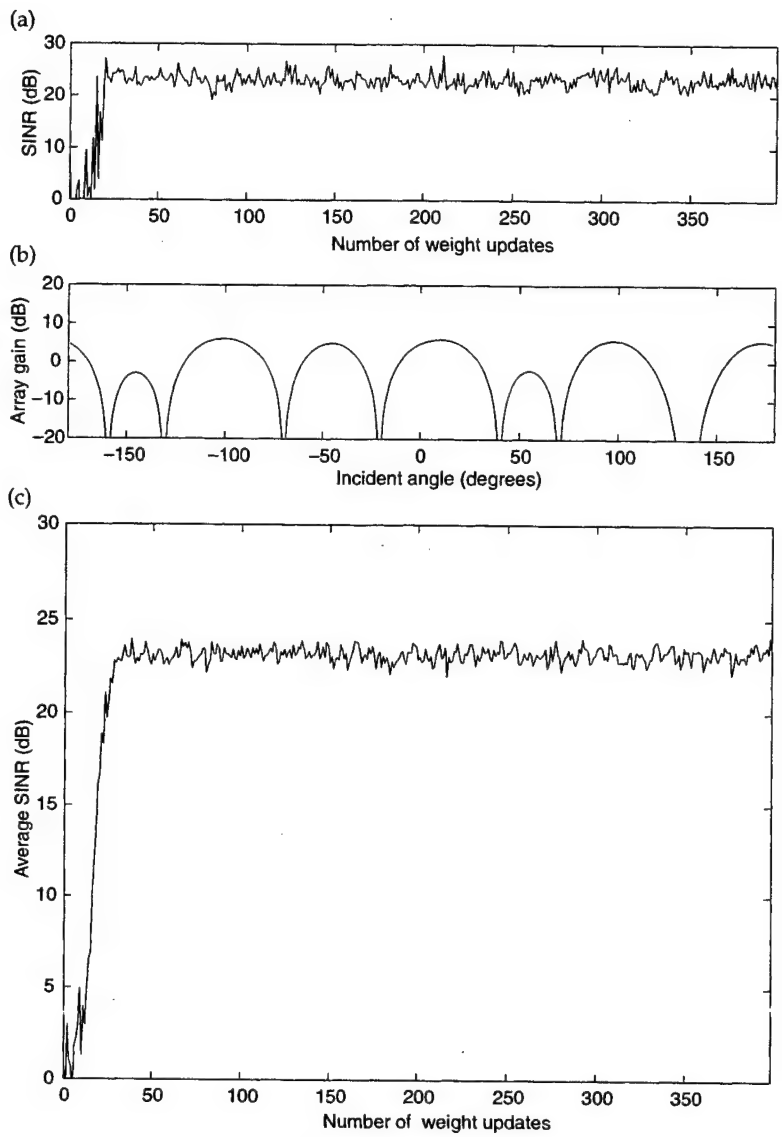


Figure 5(b) depicts the array gain pattern at the end of the simulation trial. There are nulls of less than –20 dB in the directions of the interference sources. Figure 5(c) shows the *average SINR*, which is defined as the SINR obtained by averaging over 20 simulation trials at each weight iteration.

If each interference signal independently distributes tones of equal power in every fourth frequency channel to cover 25 percent of the hopping band, then the simulation results are quite different. As shown in figures 6(a) and (c), the convergence is slowed and the fluctuations in SINR are increased, because the monitor filter only observes each interference signal 25 percent of the time. Between these observations, the weights tend to slowly drift toward the values they would have without the interference. This drift and the occasional large interference levels force the lowering of the adaptation constant to obtain good algorithm convergence, and $\alpha = 0.05$ is used to generate figure 6. Figure 6(b)

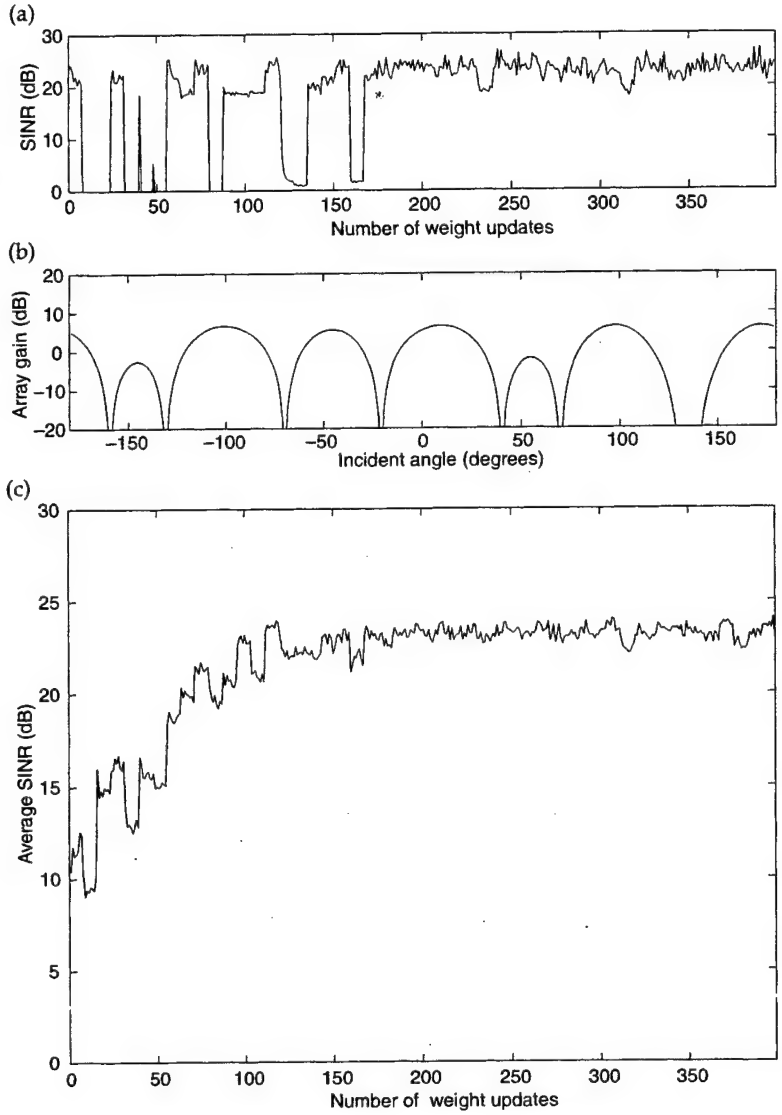


Figure 6. (a) SINR for single simulation trial with two interference signals, each comprising tones in 25 percent of frequency channels, and hopping bandwidth = 30 MHz, (b) array gain pattern at end of trial, and (c) average SINR for 20 trials.

indicates that the array gain pattern is very similar to that of the preceding example as a nearly steady-state condition is approached.

In general, interference that occupies only a small part of the hopping band, or even frequency-hopping interference signals, can be suppressed by the maximim algorithm supplemented by an error-correcting code. If the interference passes through the monitor filter often enough, it is suppressed by the maximim algorithm; if it passes through only occasionally, then it is accommodated by the error-correcting code.

The results of seven representative simulation experiments with the edge length = λ are summarized in table 2. The interference signals distribute tones in all the frequency channels and $\alpha = 0.2$ is used. Each experiment comprises 20 trials with 50 hops and 400 weight iterations per trial. The first column gives the hopping bandwidth. The second column gives the arrival angles of 1, 2, or 3 interference signals relative to the desired-signal direction. The SINRs for the last 40 weight iterations of all the trials are

averaged to obtain the *final SINR* and are used to calculate the standard deviation of the final SINR, which are listed in the third and fourth columns of the table. The final column gives the *crossing number*, which is defined as the number of weight iterations required for the average SINR over all the trials to exceed a threshold value equal to 3 dB less than the final SINR. The crossing number provides a rough measure of the relative time required for convergence to the steady state.

The first row of table 2 shows the system response to a single interference signal. The second row, which is obtained from the same experiment that gave figure 5, indicates that there is slower convergence and a 2.5-dB loss in final SINR due to the presence of a second equal-power interference signal. The third and fourth rows illustrate the further losses that occur when three equal-power interference signals are present. In the third row, the third interference signal at an angle –60° fortuitously arrives near a grating null naturally generated when the first two interference signals are nulled, as observed in figure 5(b). In the fourth row, the final SINR is much smaller because the third interference signal arrives in the natural grating lobe. The fifth and sixth rows illustrate the decrease in the final SINR and the sharp increase in the standard deviation that result when the bandwidth increases by a factor of 10 while the total interference power remains the same. The final row illustrates the limitations imposed by the resolution of the array.

The *resolution* of an adaptive array is the minimum angular separation between the interference and desired-signal sources that can be accommodated without significant performance degradation. The resolution is limited by the angular extent of a null formed against an interference signal, and is proportional to λ/D, where D is the aperture or separation of the antennas. Thus, to decrease the resolution, the edge length of the square array can be increased. However, the increase in the separation between antennas inevitably causes an increase in the number of grating nulls. The tendency of the array to form grating nulls at certain angles may impede the formation of nulls against two or more interference signals. Furthermore, since a decrease in the resolution causes a decrease in the angular width of a null, there is a decrease in the fractional bandwidth that can be accommodated by the array, as explained in the next section.

Table 2. Simulation results with edge length = λ.

Bandwidth (MHz)	Direction of interference (°)	Final SINR (dB)	Standard dev. (dB)	Crossing number
30	40	25.7	1.5	5
30	40, 70	23.2	1.3	24
30	40, 70, –60	19.8	1.9	88
30	40, 70, –50	15.5	2.1	207
300	40	23.7	5.2	6
300	40, 70	20.4	4.3	22
30	10	19.0	1.3	128

Table 3 indicates the effects of an increase in the array aperture or edge length of the square array. The superior resolution of the larger array yields greatly improved performance when the interference is from approximately the same direction as the desired signal. However, as indicated by the sixth row in the table, the performance advantage usually disappears when the hopping bandwidth is large and there are two or more equal-power interference signals.

Table 3. Simulation results with edge length = 2λ.

Bandwidth (MHz)	Direction of interference (°)	Final SINR (dB)	Standard dev. (dB)	Crossing number
30	40	25.3	1.5	2
30	40, 70	24.5	2.6	15
30	40, 70, –60	19.9	2.0	58
30	40, 70, –50	23.6	2.7	22
300	40	23.8	2.2	3
300	40, 70	14.0	8.6	12
30	10	23.9	1.3	9

6. Frequency Compensation

Adaptive processing may be significantly impaired if the *fractional bandwidth,* which is defined as the hopping bandwidth divided by the center frequency of the hopping band, exceeds a few percent and no frequency compensation is used. As an example, the hopping bandwidth in the previous example of figure 5 is expanded to 300 MHz, a 10 percent fractional bandwidth, but all the other conditions remain the same. The simulation results are shown in figure 7. The significantly increased SINR fluctuations are caused by the large changes in the carrier frequency of the frequency-hopping desired signal that sometimes occur. Not only is convergence slowed, but also the steady-state SINR is reduced relative to the results in figure 5.

Frequency compensation is sometimes needed because the antenna elements produce only relative phase information. Consider two antenna

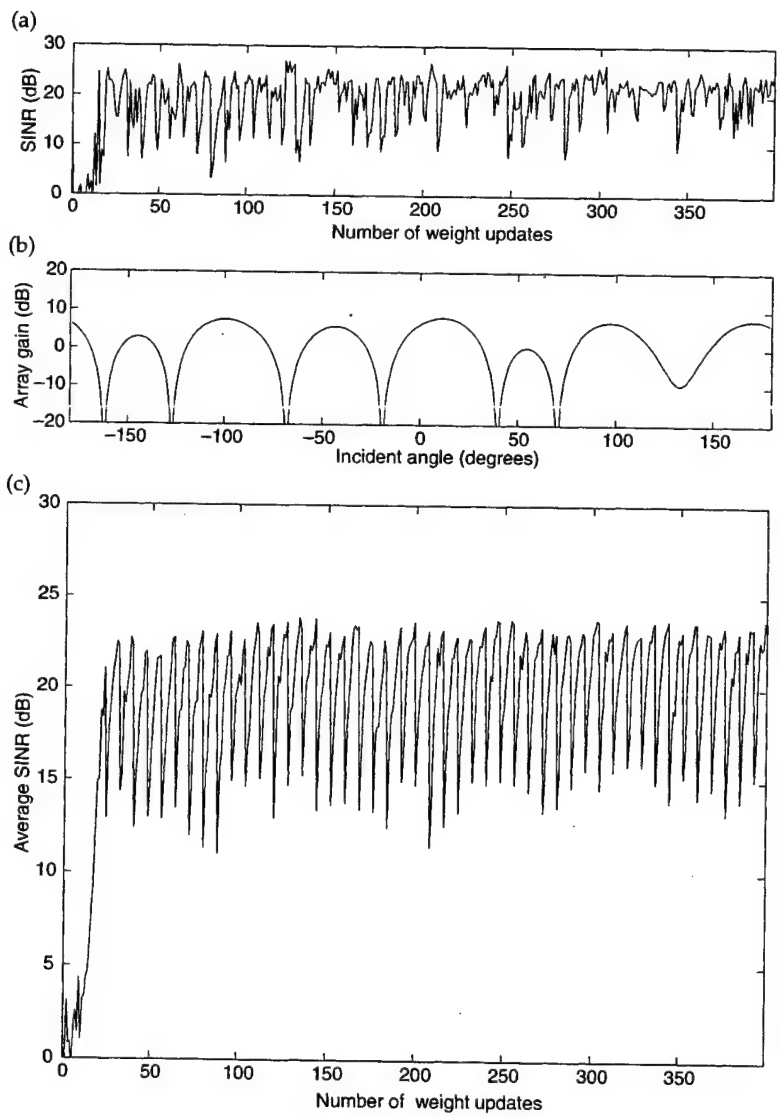


Figure 7. (a) SINR for single simulation trial with two interference signals, each comprising tones in every frequency channel, and hopping bandwidth = 300 MHz, (b) array gain pattern at end of trial, and (c) average SINR for 20 trials.

elements of an adaptive array receiving a plane wave, as illustrated in figure 8. The phase shift of a narrowband signal at one element relative to the phase of the signal at the other is

$$\theta = \frac{\omega}{c} D \sin \phi \ , \tag{63}$$

where w is the carrier frequency in radians per second, c is the velocity of an electromagnetic wave, ϕ is the arrival angle of the plane wave relative to a line perpendicular to the straight line through the two elements, and D is the distance between the elements. If the carrier frequency of the received signal changes from ω to $\omega + \Delta\omega$, then the phase shift changes by

$$(\Delta\theta)_1 = \frac{\Delta\omega}{c} D \sin \phi \ . \tag{64}$$

If the frequency is constant, but the arrival angle changes from ϕ to $\phi + \Delta\phi$, then the phase-shift change is

$$(\Delta\theta)_2 = \frac{\omega}{c} D \sin\left(\phi + \Delta\phi\right) - \frac{\omega}{c} D \sin \phi \ . \tag{65}$$

If the adaptation is realized by the use of a single adaptive weight behind each antenna element, the adaptive system cannot distinguish between the two possible causes for a change in the phase shift. Thus, a change in frequency due to frequency hopping is processed as if a change occurred in the arrival angle.

Setting $(\Delta\theta)_1$ equal to $(\Delta\theta)_2$, we conclude that the equivalent changes in frequency and arrival angle are related by

$$\sin\left(\phi + \Delta\phi\right) = \left(\frac{\Delta\omega}{\omega} + 1\right) \sin \phi \ . \tag{66}$$

This equation can be solved for $\Delta\phi$ as a function of $\Delta\omega$ if $|\sin \phi| \leq \omega/(\omega + \Delta\omega)$. For small or moderate values of $\Delta\phi$, a Taylor-series expansion of equation (66) yields

$$\Delta\phi \simeq \frac{\Delta\omega}{\omega} \tan \phi, \quad \left|\Delta\phi \tan \phi\right| << 1 \ . \tag{67}$$

Suppose that an adaptive system creates an approximate spatial null in the direction of a source of interference at the instantaneous hopping frequency ω. If the instantaneous frequency of the desired signal changes to $\omega + \Delta\omega$ and the interference source produces significant power at $\omega + \Delta\omega$, then the equivalent $\Delta\phi$ for the interference is given by

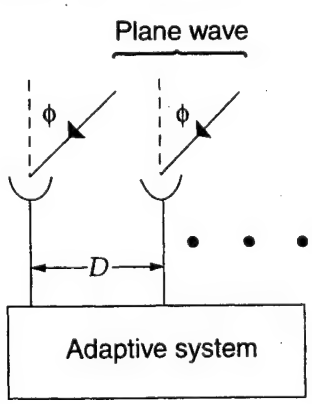


Figure 8. Two antenna elements of adaptive array receiving plane wave.

equations (66) and (67). If $\Delta\omega$ is sufficiently large, $\Delta\phi$ may be larger than the angular width of the original null. Consequently, after the carrier frequency of the desired signal hops to $\omega + \Delta\omega$, the interference is not immediately nulled, but further adaptation is required to again establish a spatial null. Thus, if the fractional bandwidth is large, then unless (1) the angular width of a null is large, (2) $\Delta\omega$ is restricted to small values, or (3) some type of frequency compensation is adopted, the benefit of the adaptive processing may diminish significantly.

The angular width of a null can be increased by decreasing the separation among antennas in the array, but then the resolution may be increased to an unsatisfactory level. The restriction of the frequency change after a hop $\Delta\omega$ to a small value keeps the interference near the center of a spatial null and makes it possible for the adaptive system to rapidly increase the interference rejection. However, the variety of hopping patterns is reduced, which diminishes the resistance to frequency-selective fading and increases the susceptibility to wideband interference or jamming. Thus, frequency compensation is the preferred alternative.

Three basic methods of frequency compensation for frequency hopping are parameter-dependent, spectral, and anticipative processing. *Parameter-dependent processing* introduces additional parameters in an attempt to increase the angular extent and spectral bandwidth of the nulls. For the maximin algorithm, a compatible realization applies v successive discrete-time outputs of each baseband filter and each monitor filter as inputs to the adaptive filter, thereby increasing the number of weights in figure 3 by a factor of v. However, the filters sharply limit the degree of spectral or beam shaping that might be possible using the successive outputs. Simulation results confirm that parameter-dependent processing is ineffective for the maximin algorithm.

Spectral processing is based on dividing the hopping band into a number of spectral regions and adapting independently when the carrier frequency is in one of the regions. The weights associated with each spectral region are stored in a memory within the weight processor of figure 3. At the end of the signal dwell time at a specific frequency, the weights of the adaptive filter are transferred to the memory. Then the weights associated with the new carrier frequency are transferred from the memory to the filter. These weights are updated during the time interval at the new frequency. Transfers between the memory and the filter are controlled by the frequency-hopping code generator. The number of spectral regions N_s is ultimately limited by the proportional increase in the required number of iterations to converge to steady state, because only $1/N_s$ of the frequency-hopping pulses are associated with each region. However, this increase is slowed by the contraction of each region as N_s increases, which expedites the convergence within each region. The relatively slow convergence rate of spectral processing is its major disadvantage; its major advantage is its relatively simple implementation.

An *anticipative adaptive system* begins adaptation toward the optimal weights for a carrier frequency before that frequency is transmitted. Anticipative processing with an auxiliary processor operating in parallel with the main processor is illustrated by figure 9. A time-advanced frequency-hopping replica hops one hopping period ahead of the replica used for dehopping the desired signal. While the main adaptive filter produces the output, the auxiliary filter adapts the weights corresponding to the next hopping frequency. After each hop, the weights associated with the new hopping frequency are transferred from the auxiliary filter to the main filter. Transfers are triggered by the clock that controls the frequency-hopping carrier transitions. The weight vector in the auxiliary filter is not necessarily updated at the same times as the weight vector in the main filter. The auxiliary filter may update its weight vector during the switching times between hops, but in the simulation, the duration of the switching time is assumed to be negligible.

The desired signal is not present in the auxiliary processors because they monitor the frequency channel of the next hop. Consequently, baseband filters can be used as monitor filters to extract various estimates of the interference-plus-noise. These baseband filters allow narrower intermediate-frequency filters and, hence, a lower sampling rate in the auxiliary processor than in the main processor.

Let $\mathbf{W}_a(l)$ denote the anticipative weight vector computed by the auxiliary adaptive filter at iteration l. The recursive equation for $\mathbf{W}_a(l+1)$ is derived by modifying the maximin algorithm to allow for the absence of the desired signal in the auxiliary filter and for the measurements by the auxiliary filter. Thus, the signal term within the brackets in equation (25) is discarded, and the remaining equation is simplified. The *anticipative maximin algorithm* is

$$\mathbf{W}_a(l+1) = \mathbf{W}_a(l) - \frac{\alpha_1 \left[\overline{\mathbf{n}^* y_{nR}}(l)\right]_a}{\hat{P}_s(l)} , \tag{68}$$

where α_1 is the *anticipative adaptation constant* and the subscript a on the bracketed factor indicates that it is measured by the auxiliary adaptive

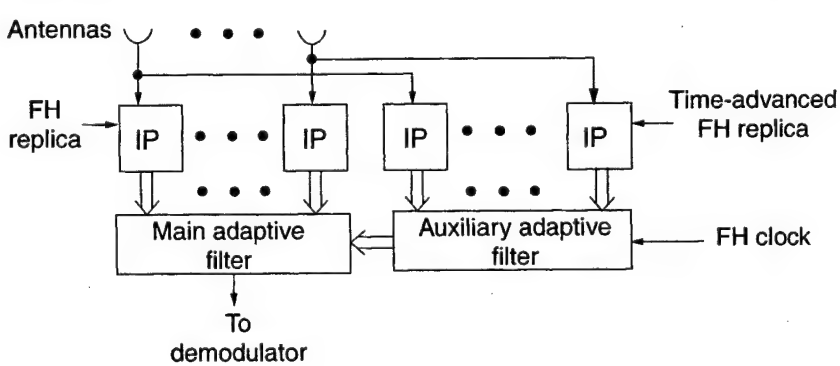


Figure 9. Basic configuration of adaptive array with anticipative processing.

filter, which uses the output of a frequency channel that will not be observed by the main adaptive filter until the next hop. The desired-signal power estimate $\hat{P}_s(l)$ is the most recent estimate computed by the main adaptive filter and is assumed to be nearly independent of the frequency channel. The bracketed factor is defined by equation (31), with $\hat{\mathbf{n}}(i)$ interpreted as the discrete-time vector of complex envelopes at the input of the auxiliary adaptive filter and $\hat{y}_{nR}(i)$ as the real part of the filter output.

The weight vector in the main adaptive filter is updated by computing equation (25), except at sampling instants occurring during switching times. At these instants, the weight vector in the main adaptive filter is set equal to that of the auxiliary adaptive filter. The switching times occur when $k = nk_0$ in the main filter and $i = nk_1$ in the auxiliary filter, where k_0 is the number of iterations per hop in the main filter, k_1 is the number of iterations per hop in the auxiliary filter, and $n = 1, 2, \ldots$ is the hop number. Thus, the maximin algorithm in the main filter becomes

$$\begin{aligned} \mathbf{W}(k+1) &= \mathbf{W}(k) + \frac{\alpha}{\hat{\rho}(k)}\left[\frac{\overline{\mathbf{s}^* y_{nR}}(k)}{\hat{P}_s(k)} - \frac{\overline{\mathbf{n}^* y_{nR}}(k)}{\hat{P}_n(k)}\right], \quad k+1 \neq nk_0 \\ \mathbf{W}(nk_0) &= \mathbf{W}_a(nk_1), \quad n = 0, 1, \ldots . \end{aligned} \tag{69}$$

The anticipative maximin algorithm, which is closely related to the recursive suppression algorithm [8] for direction finding, attempts to null all signals with power in the hopping band, but does not null the desired signal because it is never present in the baseband filters of the auxiliary adaptive filter. This absence ultimately is due to the system's knowledge of the frequency-hopping pattern.

One might consider a version of the anticipative maximin algorithm in which the signal term in equation (25) is retained. The signal term cannot be computed by the auxiliary adaptive filter because it does not receive the desired signal. Therefore, the signal term computed by the main adaptive filter is used to enable beamforming by the auxiliary adaptive filter in the direction of the desired signal. However, the signal term does not include frequency compensation, and any beamforming tends to erode the nulls when interference is absent in the monitor filter outputs. Simulation results indicate that this algorithm version can accommodate larger adaptation constants and, hence, expedites the convergence when the interference signals occupy all the frequency channels, but its performance is inferior when interference signals occupy only a fraction of the hopping band. Thus, equation (68) is the preferred form of the anticipative maximin algorithm.

The results of nine simulation experiments are displayed in table 4, where the edge length is λ, the hopping bandwidth is 300 MHz, and the fractional bandwidth is 10 percent. The first three rows in table 4 give results for no frequency compensation and $\alpha = 0.2$ as a reference for comparison.

The next three rows indicate that spectral processing with three regions and $\alpha = 0.2$ provides an increased final SINR at the cost of a large increase in the crossing number. The standard deviation is substantially reduced for one or two interference signals. Using two regions is slightly less effective. Further gain in the final SINR and a reduction in the standard deviation occurs as the number of hops is increased beyond the 50 used in generating table 4, because the weights of each of the three spectral regions approach steady-state values. This additional gain increases with the number of interference signals.

In the simulation experiments with anticipative processing, the initial value of the auxiliary weight vector is $\mathbf{W}_a(0) = [1\ 1\ 1\ 1]$, and the number of iterations per hop is $k_1 = k_0 = 8$. For one or two interference signals, the adaptation constants are $\alpha = \alpha_1 = 0.08$; for three interference signals, $\alpha = \alpha_1 = 0.02$. Many other combinations of k_1, α, and α_1 were tried, but none gave an improved overall performance compared with the results listed in the last three rows of table 4. The results indicate that anticipative processing provides better performance in both the final SINR and the standard deviation than spectral processing or no compensation, although the convergence rate of anticipative processing against three interference signals is very slow.

The simulation results indicate that when the fractional bandwidth is 10 percent, either spectral or anticipative processing usually provides a significant benefit. Anticipative processing has the advantage of usually providing the best performance, but spectral processing is simpler to implement.

Table 4. Simulation results with edge length = λ and total hopping bandwidth = 300 MHz.

Compensation method	Type	Direction of interference (°)	Final SINR (dB)	Standard dev. (dB)	Crossing number
None	—	40	23.7	5.2	6
None	—	40, 70	20.4	4.3	22
None	—	40, 70, –60	13.6	5.6	63
Spectral	$N_s = 3$	40	25.1	2.7	6
Spectral	$N_s = 3$	40, 70	22.1	2.4	70
Spectral	$k_1 = 8$	40, 70, –60	14.9	5.8	142
Anticipative	$k_1 = 8$	40	26.0	1.3	12
Anticipative	$k_1 = 8$	40, 70	23.4	1.4	32
Anticipative	$k_1 = 8$	40, 70, –60	16.0	5.1	190

7. Summary

The maximin algorithm is a blind adaptive algorithm that simultaneously forms a beam in the desired-signal direction and cancels interference in a frequency-hopping communication system. Each weight-vector iteration of the maximin algorithm is specified by the equations

$$\mathbf{W}(k+1) = \mathbf{W}(k) + \frac{\alpha}{\hat{\rho}(k)} \left[\frac{\overline{\mathbf{s}^* y_{SR}}(k)}{\hat{P}_s(k)} - \frac{\overline{\mathbf{n}^* y_{nR}}(k)}{\hat{P}_n(k)} \right] ,$$

$$\hat{\rho}(k) = \frac{\hat{P}_s(k)}{\hat{P}_n(k)} ,$$

$$\overline{\mathbf{s}^* y_{SR}}(k) = \frac{1}{m} \sum_{i=(k-1)m+1}^{km} \mathbf{X}^*(i)\, y_R(i) ,$$

$$\overline{\mathbf{n}^* y_{nR}}(k) = \frac{1}{m} \sum_{i=(k-1)m+1}^{km} \hat{\mathbf{n}}^*(i)\, \hat{y}_{nR}(i) ,$$

$$\hat{P}_n(k) = \frac{1}{m} \sum_{i=(k-1)m+1}^{km} \hat{y}_{nR}^2(i) ,$$

$$\hat{P}_s(k) = \frac{1}{m} \sum_{i=(k-1)m+1}^{km} y_R^2(i) .$$

The adaptation constant α usually should be chosen from within the range defined by

$$0 < \alpha \le 2 \left| \mathbf{W}(0) \right|^2 \rho_1 .$$

In these equations,

k = weight iteration number,
i = sample number,
m = number of samples between weight iterations,
N = number of antennas in array,
$\mathbf{X}(i)$ = input vector of baseband filter,
$y_R(i)$ = real part of output of baseband filter,
$\hat{\mathbf{n}}(i)$ = input vector of monitor filter,
$\hat{y}_{nR}(i)$ = real part of output of monitor filter, and
ρ_1 = SINR per frequency channel at an antenna input.

If the fractional bandwidths exceeds a few percent, either spectral processing or anticipative processing is effective as a means of frequency compensation. Spectral processing divides the hopping band into spectral

regions and independently adapts the weights associated with each region. The anticipative maximin algorithm for the auxiliary weight vector is

$$\mathbf{W}_a(l+1) = \mathbf{W}_a(l) - \frac{\alpha_1 \left[\overline{\mathbf{n}^* y_{nR}}(l)\right]_a}{\hat{P}_s(l)} ,$$

where the bracketed factor is measured by the auxiliary adaptive filter. The maximin algorithm in the main adaptive filter is modified by requiring that

$$\mathbf{W}(nk_0) = \mathbf{W}_a(n\,k_1), \quad n = 0, 1, \ldots .$$

In these equations,

l = anticipative weight iteration number,
α_1 = anticipative adaptation constant,
k_0 = number of weight iterations per hop in the main filter,
k_1 = number of weight iterations per hop in the auxiliary filter, and
n = hop number.

References

1. K. Bakhru and D. Torrieri, "The maximin algorithm for adaptive arrays and frequency-hopping communications," *IEEE Trans. Antenna Propag.*, vol. 32, pp. 919–928, September 1984.

2. D. Torrieri and K. Bakhru, "Frequency compensation in an adaptive antenna system for frequency-hopping communications," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 23, pp. 448–467, July 1987.

3. D. Torrieri and K. Bakhru, "An anticipative adaptive array for frequency-hopping communications," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 24, pp. 449–457, July 1988.

4. D. Torrieri, *Principles of Secure Communication Systems*, 2nd ed. Boston: Artech House, 1992.

5. J. G. Proakis, *Digitial Communications*, 3rd ed. New York: McGraw-Hill, 1995.

6. R. T. Compton, *Adaptive Antennas: Concepts and Performance*. New York: Prentice-Hall, 1988.

7. B. Porat, *A Course in Digital Signal Processing*. New York: Wiley, 1997.

8. D. Torrieri and K. Bakhru, "The Effects of Nonuniform and Correlated Noise on Superresolution Algorithms," *IEEE Trans. Antennas Propag.*, vol. 45, pp. 1214–1218, August 1997.

Distribution

Admnstr
Defns Techl Info Ctr
Attn DTIC-OCP
8725 John J Kingman Rd Ste 0944
FT Belvoir VA 22060-6218

Cubic Defense Systems
Attn K Bakhru (25 copies)
9323 Balboa Avenue
San Diego CA 92123

University of Maryland
Dept. of Electrical Engineering
Attn E Geraniotis
College Park MD 20742

Motorola SSTG
Attn MS H1175 S Chuprun (4 copies)
8201 E. McDowell Rd
Scottsdale AZ 85352

Telcordia Technologies, Inc.
Advanced Wireless Technologies Research
Attn J Liberti
331 Newman Springs Rd
Red Bank NJ 07701

US Army Rsrch Lab
Attn AMSRL-CI-AS Mail & Records Mgmt
Attn AMSRL-CI-AT Techl Pub (3 copies)
Attn AMSRL-CI-LL Techl Lib (3 copies)
Attn AMSRL-IS J D Gantt
Attn AMSRL-IS-TA J Gowens
Attn AMSRL-IS-TP D Torrieri (25 copies)
Attn AMSRL-IS-TP R Tobin
Adelphi MD 20783-1197

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE
December 1999

3. REPORT TYPE AND DATES COVERED
Final, FY99

4. TITLE AND SUBTITLE The Maximin Algorithm for Adaptive Arrays and Frequency-Hopping Systems

5. FUNDING NUMBERS
DA PR: NA
PE: NA

6. AUTHOR(S) Don Torrieri (ARL), Kesh Bakhru (Cubic Defense Systems)

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
U.S. Army Research Laboratory
Attn: AMSRL-IS-T email: dtorrieri@arl.mil
2800 Powder Mill Road
Adelphi, MD 20783-1197

8. PERFORMING ORGANIZATION REPORT NUMBER
ARL-TR-2026

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
U.S. Army Research Laboratory
2800 Powder Mill Road
Adelphi, MD 20783-1197

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES
ARL PR: NA
AMS code: NA

12a. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

The maximin algorithm adapts the weights of an antenna array to provide simultaneous interference suppression and beamforming in a frequency-hopping communication system. The algorithm is derived and a digital implementation is presented. Simulation experiments illustrate the ability of the maximin algorithm to rapidly form deep nulls in the directions of interference sources. For the full exploitation of an adaptive array for wideband frequency-hopping communications, some form of frequency compensation is necessary. The strengths, limitations, and variations of three methods are examined.

14. SUBJECT TERMS
Adaptive arrays, frequency hopping, maximin algorithm

15. NUMBER OF PAGES
37

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-18
298-102